High Assurance Computer Systems: A Research Agenda

John McLean and Constance Heitmeyer
Center for High Assurance Computer Systems
Naval Research Laboratory
Washington, DC 20375

1 Introduction

As computers and their supporting communication networks have become increasingly enmeshed in our national technological fabric, we have become increasingly dependent on *high assurance computer systems*, i.e., computer systems for which compelling evidence is required that the system delivers its services in a manner that satisfies certain critical properties. Obvious examples of high assurance systems include military systems (e.g., weapon systems, C^4I systems, etc), flight programs for both commercial and military aircraft, air traffic control systems, financial and commerce systems, medical systems (including medical databases and medical equipment), etc. Less obvious examples are the various components of the information infrastructure that supports such systems and their communications (e.g., the NII).

These systems are extremely complicated and the science and engineering principles that underlie them are yet to be completely worked out. Nevertheless, our national well-being depends upon these systems satisfying certain *critical properties* including:

- **security properties**, which prevent unauthorized disclosure, modification, and withholding of sensitive information, even when under attack by a hostile agent;
- **safety properties**, which prevent unintended events that result in death, injury, illness, or damage to or loss of property;

- **real-time properties**, which require the system to deliver its results within specified time intervals; and

- **fault-tolerance properties**, which require the system to guarantee a certain quality of service despite faults, such as hardware, workload, or environmental anomalies.

In recent years, many scientific approaches for specifying, constructing, and certifying high assurance systems have been developed, including formal specification techniques, formal models, design methods, and rigorous verification and validation techniques. Much still needs to be done with respect to individual critical properties, but even more needs to be done with respect to two difficult problems that have hardly been addressed. The first is the lack of technology to support the application of these new techniques and methods to practical, real-world systems. Without such technology, opportunities to transfer many of the basic research results to industry are severely limited.

The second problem is the lack of a unified framework for building systems that satisfy several critical properties. This problem arises because not one but four different approaches for developing high assurance systems (one for each property identified above) have evolved. Each approach has a different overall philosophy of system development and different techniques and methods for specification and assurance. None of the four separate approaches, by itself, is sufficient to handle systems now being built which must satisfy two or more critical properties simultaneously. A scientific basis (i.e., formal specification techniques, formal models, assurance methods, etc.) is needed for constructing systems that must be simultaneously secure, safe, timely, and fault-tolerant.

These problems are of such a difficult nature that they will not be solved by industry or academia without the support of both government funding agencies and government research laboratories. In the first place, history has shown that industry shies away from tackling such critical properties, focusing instead on increasing system functionality in order to claim a share of the market place.[1] The reason for this is that industry is too short-termed

[1]One of many examples is the cellular phone industry which forsook security in the initial stages of system development in an effort to bring products to the market place as soon as possible. Security is now being addressed as companies are suffering from the

and bottom-line oriented to invest the substantial time, risk, and dollars necessary to tackle critical property issues.

Secondly, our increasing national dependence on many of these systems (e.g, the flight control system, financial systems, and commerce systems) renders them, in effect, military targets. This problem is exacerbated by the fact the Department of Defense, itself, has come to rely on non-DoD systems. The need to make use of COTS products and the need to form a complete tactical picture using both information gathered from the world-wide "infosphere" and information formerly bottled-up in special TS enclaves are rapidly erasing the distinction between DoD and non-DoD system. Industry and academia alone are not capable of addressing the implications of this fact. Commercial levels of assurance are no longer sufficient for many traditionally commercial systems, and neither industry nor academia currently have sufficient expertise to increase these levels of assurance to DoD quality.

On February 21-23, 1995, a group of fifty experts from academia, government, and industry participated in the High Assurance Computing Workshop in Washington, DC, to address these challenging problems. The workshop was organized into four tracks, one for each of the four properties. Although initially the four tracks met individually, most of the remaining track discussions were between pairs of tracks. In these discussions, the current state-of-the-art and state-of-the practice for developing high assurance systems was assessed, critical issues and promising approaches that crossed technical disciplines were proposed, and obstacles to integrating the disciplines were identified as were promising avenues for overcoming these obstacles. To help focus the discussion on realistic problems, four invited talks were presented during the course of the workshop. Each talk described a high assurance system currently under development and the problems that exist in developing such systems. On the final day of the workshop, each track drafted a focused science and technology agenda identifying specific research topics. The participants were asked to choose topics with high potential payoff in areas where substantial new investments are necessary to accelerate progress.

This paper briefly summarizes the results of the workshop, identifies several examples of high assurance systems that were discussed, provides some details of one of the systems as a representative example, and describes some

loss of millions of dollars each year due to phone fraud. The issue of privacy for phone subscribers is yet to be meaningfully addressed.

problems that need to be addressed by basic research as well as problems that hinder the use of research results in practical applications. Finally, we propose a research agenda.

2 Examples of High Assurance Systems

Although the Introduction focused on high assurance systems of a national scale, high assurance systems exist on all scales. In fact, the increasing intrusion of computers and microprocessors into our daily lives has started to render such systems commonplace. Most systems developed today already must meet real-time requirements if they are to be responsive enough to fulfill their function. As systems take over more important tasks and interact with a greater number of potentially dangerous operational devices, dependability and safety become concerns. As soon as any such system is placed in a potentially hostile environment (where the "intruder" could be a dedicated agent or a bored teenager), security becomes a concern. In fact, we are heading toward a state where dependence on high-assurance systems will be the norm. Yet, we currently have little knowledge of how to build complex examples of such systems, or even simple examples of such systems if we follow the current trend of using "integrated" rather than "federated" system architectures.

A wide variety of high assurance systems were discussed during the workshop, including the security design of an avionics system, the next-generation U.S. air traffic control system, the Boeing 777 flight control program, a medical system that uses robots to prepare human bones for artificial joints, a second medical system that controls a heart defibrillator, the Joint Maritime Command Information System (JMCIS), and a "people mover" personalized transportation system. Each of these systems is required to satisfy all four of the properties identified above. This can be seen by simply considering one of the systems: the heart defibrillator system.

This medical system controls an implanted defibrillator, a device which senses and treats bradycardia (slow heart rhythms) and tachychardia (fast heart rhythms) and records information about how these conditions are detected and treated[2]. One of the critical functions performed by the device is bradycardia therapy, which delivers a low energy pacing pulse to the patient's

[2]The discussion of this system presented here draws heavily on [2].

heart when a slow beat is detected. This provides the patient with a minimum life-sustaining heart rate. The device also delivers tachycardia therapy, which delivers a high energy shock to the patient when an abnormally fast heart beat is detected. In the case of bradycardia therapy, the physician sets the peak voltage (amplitude) and the duration (width) of the pulse. In the case of tachycardia therapy, the physician programs the energy level of the shock. To set the various parameters used by the defibrillator, the physician uses another device in the system called the "programmer".

Clearly, this system is a safety-critical system. Among the hazards that can arise in such a system are:

- incorrect diagnosis based on the stored history data
- ineffective or inappropriately delivered therapy
- nondelivered therapy

In addition to operating safely, the system also has stringent real-time requirements: it must perform diagnosis and take therapeutic actions within specified time limits. Moreover, the device must clearly be fault-tolerant: because it is implanted, the device is difficult to service.

Finally, the system must satisfy security properties. The programmer device, which the physician uses to communicate with the implanted unit and through which parameters are set and therapy history is acquired and displayed, contains sensitive patient data that must be protected. Moreover, access to the device must be restricted, since operation by unqualified personnel poses a potential hazard to the patient. All of these are aspects of the required system security.

3 Problems in the Development of High Assurance Systems

One way to see why research is needed for such systems is to examine problems encountered when developing such a system. An illustrative example is the AAS, the proposed next-generation air-traffic control (ATC) system contracted by the American FAA.[3]

[3]The discussion of the AAS presented here draws heavily on [1].

The AAS was originally intended to use the relatively mature delta-T atomic broadcast technology. Although the delta-T technology seemed intuitively well-matched for the AAS needs, it had two shortcoming which had to be addressed:

1. The fundamental delta-T protocols imposed communication delays on the order of 200-300 ms for reliable message passing. This was unacceptable given AAS hard real-time constraints.

2. The delta-T technology did not provide a complete fault-tolerant methodology for building large distributed systems.

Although either one of these problems could be solved in isolation, solving both of them proved impossible. Protocols could be reduced to meet timing constraints, but not without compromising the model's fault-tolerance features.

A similar problem was encountered with the system's membership tracking protocol. The algorithm was based on a detectable "heartbeat" which processors could monitor in order to detect one-another's failures. The problem lay in trying to find a heartbeat rate that was fast enough for member processes to detect the failure of another process in a timely manner, yet was slow enough not to flood the system.

As many members of the Workshop's Dependability Track noted, the failure of the AAS effort was "a nearly inevitable outcome, because the fundamental theoretical research was not in place to solve this kind of problem [i.e., the marriage of two or more high assurance properties], and because the existing theoretical results were not reflected in a corresponding integrated, scalable, and commercially viable software infrastructure." [1]

Extrapolating from the inability to achieve hard real-time constraints and fault-tolerance in a single system, it is clear that the basic research is also lacking to achieve the integration of either of these properties and security or safety. Nobody would deny that safety is a vital concern to such a system. Given the increased threat this country faces in the form of terrorism, security is also a major concern. If flight information had to be encrypted to maintain secrecy or digitally signed for purposes of authentication, the extra processing and protocol layers that would be necessary would only exacerbate the problem of meeting the real-time constraints. In a military context,

the unpredictability that anti-jam and covert channel suppression techniques typically introduce would fly in the face of the timing predictability that is required by current real-time practices.

The problem is not so much that we do not know how to achieve any of these properties individually (although, as we shall discuss below, this is a problem). The harder problem is that we do not know how to achieve one property without undercutting our ability to achieve a different, yet equally important, property. Further, certain properties are dependent upon each other. For example, in a military aircraft, information that is necessary to maintain for access control information must be protected by fault-tolerance techniques. Yet these techniques must be protected by access controls from being subverted. Layering these mutually dependent properties in a workable way is currently an art without an underlying foundation.

4 Research Agenda

The size of current and future computer systems necessitates a modular approach to system development. However, such an approach assumes that we have an adequate understanding of how to break a system down into modules, of the system modules themselves, and of how these modules will interact. Although our understanding of this process is probably close to being in-hand for functional properties, this is not true for the properties that we have described in this paper. Security properties are notorious for being preserved by neither refinement nor composition.[3] Knowledge of the security properties of a system's components and of the connections among these components in the system is far from sufficient for determining the security properties of the aggregate system made up from those components. Although the security community is beginning to make some headway with these problems, there is still a great deal of work to be done.[4] The stochastic nature of many timing properties and the "emergent" nature of safety properties renders composition and refinement problematic for them as well. The fault-tolerance community is just starting to realize that they also face similar problems. The result is that we have an insufficient engineering methodology

[4]In fact, the current trend embodied in the Multi-level Information System Security Initiative (MISSI) of attempting to achieve security via a small set of profiled, relatively low-assurance components is a great step backwards from this point of view.

for constructing systems that satisfy even a single critical property.

This observation leads to the first pillar of our research agenda:

- **Initiate a research program (basic through applied) for the study of critical system properties (fault tolerance, real-time, safety, and security) and methods for building systems that satisfy these properties.**

When we turn to systems that must satisfy multiple such properties, things get substantially worse. As seen in the discussion of the AAS example, current techniques for achieving one critical property often undercuts our ability to achieve a second critical property. Even when such undercutting does not occur, it is still an open issue how to best layer critical properties that support each other. One reason for this is that we have no common framework for representing and reasoning about multiple critical properties.

This leads to the second pillar of our research agenda:

- **Initiate a research program (basic through applied) to examine the interaction of critical properties and methods for building systems that must satisfy multiple critical properties.**

This second pillar will have the additional bonus that it will increase cross-community communication. One fact that emerged from the High Assurance Computing Workshop was that techniques developed for one critical property could sometimes be used to support another critical property. Hence, during a recent joint meeting of IFIP 10.4 and 11.3 working groups, the fault-tolerance working group was extremely interested in the composition research performed by the security working group. In a related vein, the safety community is starting to explore the use of security kernels for enforcing safety properties.

However, getting research out of local research communities into the wider research community is insufficient. We need also to lead this research into the development community. To do this we shall need to develop industry quality supporting tools to apply existing research methods on real systems, and we shall need a few industrial-sized systems to serve as a example applications for these research methods and their supporting tools.

This leads to the third pillar of our research agenda:

- **Initiate a program focused on building the tools necessary to support the application of research methods to real systems and experimental developments of systems that must satisfy one or more critical property.**

This third pillar points to a final objective that must be met if research in this area is to succeed. Large, high assurance systems must be built by industry. However, industry lacks the capability to perform the necessary research into how to do this. Academia alone cannot provide this research since much of the expertise about how to obtain sufficiently high levels of assurance for many of these critical properties resides on government research laboratories.

This leads to the fourth pillar of our research agenda:

- **Initiate a program focused on building collaboration between industry, academia, and government research laboratories.**

It should be stressed that none of these research pillars can solve the problem we face with respect to high-assurance systems by itself. Nor will dumping a large amount of money for a short period of time at the problem lead to a solution. Systems will continue to increase in complexity, and solutions developed today will not work for tomorrow's systems. What is needed is a recognition of the problem and a long-termed commitment to address the problem today and continue addressing the problem in the future. A modest investment over 20 years would be much more productive than a major investment over 5 years. A long term commitment in all four research areas is what is needed. Anything short of such a commitment will eventually lead to a national disaster.

Acknowledgements

We wish to thank Gary Koob, Carl Landwehr, Teresa Lunt, and John Rushby for their assistance in organizing the Workshop on High Assurance Computing, the workshop track chairs – Ricky Butler, John Knight, Nancy Lynch, and Catherine Meadows – and the workshop participants.

References

[1] Ricky Butler, et. al., *Workshop on High Assurance Computing: Dependability Track Report,* Workshop on High Assurance Computing, Washington, DC, Feb. 21-23, 1995.

[2] John Knight, et. al., *Workshop on High Assurance Computing: Safety Track Report,* Workshop on High Assurance Computing, Washington, DC, Feb. 21-23, 1995.

[3] John McLean, "Security Models," in *Encyclopedia of Software Engineering*, ed. John Marciniak, Wiley Press, 1994.

[4] Catherine Meadows, et. al., *Workshop on High Assurance Computing: Security Track Report,* Workshop on High Assurance Computing, Washington, DC, Feb. 21-23, 1995.